****

**SAFETY HELMET DETECTION SYSTEM USING**

**DEEP LEARNING TECHNIQUES**

**A PROJECT REPORT**

***Submitted by***

**DINESSH RAIJ.V (811719104024)**

**GOKUL.P (811719104029)**

**GOKUL RAJ .R (811719104030)**

**NISHAANTH.S (811719104061)**

***in partial fulfillment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

***in***

**COMPUTER SCIENCE ENGINEERING**

**K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY**

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

**SAMAYAPURAM – 621 112**

**JUNE, 2022.**

**K.RAMAKRISHNAN COLLEGE OF TECHNOLOGY**

**(Autonomous)**

**SAMAYAPURAM – 621 112**

**BONAFIDE CERTIFICATE**

Certified that this project report titled **“SAFETY HELMET DETECTION SYSTEM USING DEEP LEARNING TECHNIQUES”** is the bonafide work of **DINESSH RAJH.V (811719104024), GOKUL.P (811719104029), GOKUL RAJ.R (811719104030), NISHANTH.S (811719104061)** who carried out the project under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

|  |  |
| --- | --- |
| **SIGNATURE**  Mr. M.SIVAKUMAR M.E., (Ph.D).,  **HEAD OF THE DEPARTMENT**  Department of CSE  K.Ramakrishnan College of Technology  (Autonomous)  Samayapuram – 621 112. | **SIGNATURE**  **SUPERVISOR**  ASSISTANT PROFESSOR  Department of CSE  K.Ramakrishnan College of Technology  (Autonomous)  Samayapuram – 621 112. |
|  |  |

Submitted for the viva-voce examination held on ………………

**Internal Examiner External Examiner**

**DECLARATION**

We jointly declare that the project report on **“SAFETY HELMET DETECTION SYSTEM USING DEEP LEARNING TECHNIQUES”** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **“ANNA UNIVERSITY CHENNAI”** for the requirement of Degree of **BACHELOR OF ENGINEERING**. This project report is submitted on the partial fulfilment of the requirement of the award of Degree of **BACHELOR OF ENGINEERING**.

|  |
| --- |
| **Signature** |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  DINESSH RAJH.V |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  GOKUL.P  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  GOKUL RAJ.R  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  NISHAANTH.S |
|  |
|  |

Place: Samayapuram

Date:

**ACKNOWLEDGEMENT**

It is with great pride that we express our gratitude and in-debt to our institution “**K.Ramakrishnan College of Technology (Autonomous)**”, for providing us with the opportunity to do this project.

We are glad to credit honourable chairman **Dr. K.RAMAKRISHNAN**, **B.E.,** for having provided for the facilities during the course of our study in college.

We would like to express our sincere thanks to our beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to our project and offering adequate duration in completing our project.

We would like to thank **Dr. N. VASUDEVAN, M.E., Ph.D.,** Principal, who gave opportunity to frame the project the full satisfaction.

We whole heartily thanks to **Mr. M. SIVAKUMAR**, **M.E., (Ph.D).,** Head of the department, **COMPUTER SCIENCE ENGINEERING** for providing his encourage pursuing this project.

I express my deep and sincere gratitude to my project guide **Mrs.** **SILABARASI, M.TECH.,** Department of **COMPUTER SCIENCE ENGINEERING,** for his incalculable suggestions, creativity, assistance andpatience which motivated me to carry out this project.

I render my sincere thanks to Course Coordinator and other staff members for providing valuable information during the course.

I wish to express my special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

**TABLE OF CONTENTS**

**CHAPTER TITLE PAGE NO.**

**LIST OF TABLES 5**

**ABSTRACT 6**

1. **INTRODUCTION 8**

1.1 artificial intelligence 9

1.2 deep learning  11

1. **LITERATURE SURVEY 13**

2.1 Analaysis of smart helmets and Designing an IOT based

on smart helmet 13

2.2 SSD 13

2.3Faster R-CNN 14

2.4 Computer Vision and Pattern Recognition 15

1. **EXISTING SYSTEM 16**

3.1 Drawback of existing system 16

3.2 Proposed system 17

3.3 Advantages of proposed system 19

3.4 Propsed methodology 21

1. **CONVOLUTIONAL NEURAL NETWORK 22**

4.1 Convoutional layers 23

4.2 Different layers of CNN 26

**5 SOFTWARE DISCRIPTION 29**

5.1 Python 3.7.3 30

5.2 Tensorflow 41

5.3 Keras 43

5.4 Numpy 50

5.5 Pillow 52

**6 RESULTS 67**

6.1 Conclusions 69

**SAFETY HELMET DETECTION SYSTEM USING DEEP LEARNING TECHNIQUES**

**ABSTRACT**

Automatically detect whether workers are wearing safety helmet at construction site is significant for safety production. Concerning the problem that the existing safety helmet wearing detection method is difficult to detect the partial occlusion, different size and small object, and the detection accuracy is low. In this paper, we present an advanced deep learning based approach to determine whether workers are wearing safety helmets. In our framework, we first use the multi-scale training and the increasing anchors strategies to enhance the robustness of the original Faster RCNN algorithm to detect different scales and small object. This paper presents a deep learning approach for accurate safety helmets wearing detection in employing a single shot multi-box detector. Moreover, because of safety helmet usually relatively small and unfortunately SSD struggles in detecting very small objects, a novel and practical safety helmet wearing detecting system is proposed, Finally, extensive compelling experimental results in construction working site illustrate the efficiency and effectiveness of our work.

**INTRODUCTION**

Construction is a high-risk industry where construction workers tend to be hurt in the work process. Head injuries are very serious and often fatal. According to the accident statistics released by the state administration of work safety from 2015 to 2018, among the recorded 78 construction accidents, 53 events happened owing to the fact that the workers did not wear safety helmets properly, accounting for 67.95% of the total number of accidents .

In safety management at the construction site, it is essential to supervise the safety protective equipment wearing condition of the construction workers. Safety helmets can bear and disperse the hit of falling objects and alleviate the damage of workers falling from heights. Construction workers tend to ignore safety helmets because of weak safety awareness. At the construction site, workers that wear safety helmets improperly are much more likely to be injured. Traditional supervision of the workers wearing safety helmets on construction sites often requires manual work [2]. There are problems such as a wide range of operations and difficult management of site workers. These factors make manual supervision difficult and inefficient and it is difficult to track and manage the whole workers at the construction sites accurately in real time [3]. Hence, it is hard to satisfy the modern requirement of construction safety management only relying on the traditional manual supervision. In this context, it remains a significant issue to study on the automatic detection and recognition of safety helmets wearing conditions.

The automatic monitoring method can contribute to monitoring the construction workers and confirm the safety helmet wearing conditions at the construction site. In particular, considering that the traditional manual supervision of the workers is often costly, time-consuming, error-prone, and not sufficient to satisfy the modern requirements of construction safety management, the automatic supervision method can be beneficial to real-time on-site monitoring.

In this paper, based on the previous studies on computer vision-based object detection, we develop a deep learning-based method for the real-time detection of safety helmet at the construction site. The major contributions are as follows: (1) a dataset containing 3261 images of safety helmets collected from two sources, i.e., manual capture from the video monitoring system at the workplace and open images obtained using web crawler technology, is established and released to the public. (2) The Faster RCNN algorithm that is based on convolutional neural networks is used to train the model, which is verified in our study as an alternative solution to detect the unsafe operation of failure of wearing a helmet at the construction site.

**ARTIFICIAL INTELLIGENCE**

In [computer science](https://en.wikipedia.org/wiki/Computer_science), artificial intelligence (AI), sometimes called machine intelligence, is [intelligence](https://en.wikipedia.org/wiki/Intelligence) demonstrated by [machines](https://en.wikipedia.org/wiki/Machine), in contrast to the natural intelligence displayed by humans. Leading AI textbooks define the field as the study of "[intelligent agents](https://en.wikipedia.org/wiki/Intelligent_agent)": any device that perceives its environment and takes actions that maximize its chance of successfully achieving its goals.[[1]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-Definition_of_AI-1) Colloquially, the term "artificial intelligence" is often used to describe machines (or computers) that mimic "cognitive" functions that humans associate with the [human mind](https://en.wikipedia.org/wiki/Human_mind), such as "learning" and "problem solving".[[2]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-FOOTNOTERussellNorvig20092-2)

As machines become increasingly capable, tasks considered to require "intelligence" are often removed from the definition of AI, a phenomenon known as the [AI effect](https://en.wikipedia.org/wiki/AI_effect).[[3]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-3) A quip in Tesler's Theorem says "AI is whatever hasn't been done yet."[[4]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-4) For instance, [optical character recognition](https://en.wikipedia.org/wiki/Optical_character_recognition) is frequently excluded from things considered to be AI [[5]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-5) , having become a routine technology.[[6]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-6) Modern machine capabilities generally classified as AI include successfully [understanding human speech](https://en.wikipedia.org/wiki/Natural_language_understanding),[[7]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-FOOTNOTERussellNorvig2009-7) competing at the highest level in [strategic game](https://en.wikipedia.org/wiki/Strategic_game) systems (such as [chess](https://en.wikipedia.org/wiki/Chess) and [Go](https://en.wikipedia.org/wiki/Go_(game))),[[8]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-bbc-alphago-8) [autonomously operating cars](https://en.wikipedia.org/wiki/Autonomous_car), intelligent routing in [content delivery networks](https://en.wikipedia.org/wiki/Content_delivery_network), and [military simulations](https://en.wikipedia.org/wiki/Military_simulations).

Artificial intelligence was founded as an academic discipline in 1955, and in the years since has experienced several waves of optimism,[[9]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-Optimism_of_early_AI-9)[[10]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-AI_in_the_80s-10) followed by disappointment and the loss of funding (known as an "[AI winter](https://en.wikipedia.org/wiki/AI_winter)"),[[11]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-First_AI_winter-11)[[12]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-Second_AI_winter-12) followed by new approaches, success and renewed funding.[[10]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-AI_in_the_80s-10)[[13]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-AI_in_2000s-13) For most of its history, AI research has been divided into subfields that often fail to communicate with each other.[[14]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-Fragmentation_of_AI-14) These sub-fields are based on technical considerations, such as particular goals (e.g. "[robotics](https://en.wikipedia.org/wiki/Robotics)" or "[machine learning](https://en.wikipedia.org/wiki/Machine_learning)"),[[15]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-Problems_of_AI-15) the use of particular tools ("[logic](https://en.wikipedia.org/wiki/Logic)" or [artificial neural networks](https://en.wikipedia.org/wiki/Artificial_neural_network)), or deep philosophical differences.[[16]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-Biological_intelligence_vs._intelligence_in_general-16)[[17]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-Neats_vs._scruffies-17)[[18]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-Symbolic_vs._sub-symbolic-18) Subfields have also been based on social factors (particular institutions or the work of particular researchers).[[14]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-Fragmentation_of_AI-14)

The traditional problems (or goals) of AI research include [reasoning](https://en.wikipedia.org/wiki/Automated_reasoning), [knowledge representation](https://en.wikipedia.org/wiki/Knowledge_representation), [planning](https://en.wikipedia.org/wiki/Automated_planning_and_scheduling), [learning](https://en.wikipedia.org/wiki/Machine_learning), [natural language processing](https://en.wikipedia.org/wiki/Natural_language_processing), [perception](https://en.wikipedia.org/wiki/Machine_perception) and the ability to move and manipulate objects.[[15]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-Problems_of_AI-15) [General intelligence](https://en.wikipedia.org/wiki/Artificial_general_intelligence) is among the field's long-term goals.[[19]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-General_intelligence-19) Approaches include [statistical methods](https://en.wikipedia.org/wiki/Artificial_intelligence#Statistical), [computational intelligence](https://en.wikipedia.org/wiki/Artificial_intelligence#Sub-symbolic), and [traditional symbolic AI](https://en.wikipedia.org/wiki/Artificial_intelligence#Symbolic). Many tools are used in AI, including versions of [search and mathematical optimization](https://en.wikipedia.org/wiki/Artificial_intelligence#Search_and_optimization), [artificial neural networks](https://en.wikipedia.org/wiki/Artificial_neural_network), and [methods based on statistics, probability and economics](https://en.wikipedia.org/wiki/Artificial_intelligence#Probabilistic_methods_for_uncertain_reasoning). The AI field draws upon [computer science](https://en.wikipedia.org/wiki/Computer_science), [information engineering](https://en.wikipedia.org/wiki/Information_engineering_(field)), [mathematics](https://en.wikipedia.org/wiki/Mathematics), [psychology](https://en.wikipedia.org/wiki/Psychology), [linguistics](https://en.wikipedia.org/wiki/Linguistics), [philosophy](https://en.wikipedia.org/wiki/Philosophy), and many other fields.The field was founded on the assumption that [human intelligence](https://en.wikipedia.org/wiki/Human_intelligence) "can be so precisely described that a machine can be made to simulate it".[[20]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-20) This raises philosophical arguments about the nature of the [mind](https://en.wikipedia.org/wiki/Mind) and the ethics of creating artificial beings endowed with human-like intelligence. These issues have been explored by [myth](https://en.wikipedia.org/wiki/History_of_AI#AI_in_myth,_fiction_and_speculation), [fiction](https://en.wikipedia.org/wiki/Artificial_intelligence_in_fiction) and [philosophy](https://en.wikipedia.org/wiki/Philosophy_of_AI) since [antiquity](https://en.wikipedia.org/wiki/Ancient_history).[[21]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-McCorduck's_thesis-21) Some people also consider AI to be [a danger to humanity](https://en.wikipedia.org/wiki/Existential_risk) if it progresses unabated.[[22]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-22)[[23]](https://en.wikipedia.org/wiki/Artificial_intelligence#cite_note-pmid31835078-23) Others believe that AI, unlike previous technological revolutions, will create a [risk of mass unemployment](https://en.wikipedia.org/wiki/Technological_unemployment#21st_century). isa process where level of consciousness decrease due to lack of sleep or fatigue and can cause a n falls asleep. When driver is drowsy, the **DEEP LEARNING**

Deep learning (also known as deep structured learning or differential programming) is part of a broader family of [machine learning](https://en.wikipedia.org/wiki/Machine_learning) methods based on [artificial neural networks](https://en.wikipedia.org/wiki/Artificial_neural_networks) with [representation learning](https://en.wikipedia.org/wiki/Representation_learning). Learning can be [supervised](https://en.wikipedia.org/wiki/Supervised_learning), [semi-supervised](https://en.wikipedia.org/wiki/Semi-supervised_learning) or [unsupervised](https://en.wikipedia.org/wiki/Unsupervised_learning).

Deep learning architectures such as [deep neural networks](https://en.wikipedia.org/wiki/Deep_learning#Deep_neural_networks), [deep belief networks](https://en.wikipedia.org/wiki/Deep_belief_network), [recurrent neural networks](https://en.wikipedia.org/wiki/Recurrent_neural_networks) and [convolutional neural networks](https://en.wikipedia.org/wiki/Convolutional_neural_networks) have been applied to fields including [computer vision](https://en.wikipedia.org/wiki/Computer_vision), [speech recognition](https://en.wikipedia.org/wiki/Automatic_speech_recognition), [natural language processing](https://en.wikipedia.org/wiki/Natural_language_processing), [audio recognition](https://en.wikipedia.org/wiki/Audio_recognition), social network filtering, [machine translation](https://en.wikipedia.org/wiki/Machine_translation), [bioinformatics](https://en.wikipedia.org/wiki/Bioinformatics), [drug design](https://en.wikipedia.org/wiki/Drug_design), medical image analysis, material inspection and [board game](https://en.wikipedia.org/wiki/Board_game) programs, where they have produced results comparable to and in some cases surpassing human expert performance.

[Artificial neural networks](https://en.wikipedia.org/wiki/Artificial_neural_network) (ANNs) were inspired by information processing and distributed communication nodes in biological systems. ANNs have various differences from biological [brains](https://en.wikipedia.org/wiki/Brain). Specifically, neural networks tend to be static and symbolic, while the biological brain of most living organisms is dynamic (plastic) and analog. In deep learning, each level learns to transform its input data into a slightly more abstract and composite representation. In an image recognition application, the raw input may be a [matrix](https://en.wikipedia.org/wiki/Matrix_(mathematics)) of pixels; the first representational layer may abstract the pixels and encode edges; the second layer may compose and encode arrangements of edges; the third layer may encode a nose and eyes; and the fourth layer may recognize that the image contains a face. Importantly, a deep learning process can learn which features to optimally place in which level on its own. (Of course, this does not completely eliminate the need for hand-tuning; for example, varying numbers of layers and layer sizes can provide different degrees of abstraction.)

**CHAPTER 2**

**LITERATURE SURVEY**

**2.1 Analysis of Smart helmets and Designing an IoT based smart helmet: A cost effective solution for Riders**

we have reviewed the recent trends in developing Smart Helmet system. The smart helmet system is used to prevent the accidents in motor bikes and to identify the bike accidents on time for wellness of human being. Also, the smart helmet system analyzed in this paper is used in mining industry for safeguarding the miners from hazardous events in the mine and to alert the miners from hazardous gas emissions inside it. The research also helps to understand the smart helmet system evolved over the period and currently by using emerging technology like Internet of Things (IoT). This work also addresses the intelligent motor bike helmet system which is used to inform the rider about rear big trucks/buses for avoiding collisions.

**2.2.SSD: single shot multibox detector [C] Proc of European Conference on Computer Vision.**

We present a method for detecting objects in images using a single deep neural network. Our approach, named SSD, discretizes the output space of bounding boxes into a set of default boxes over different aspect ratios and scales per feature map location. At prediction time, the network generates scores for the presence of each object category in each default box and produces adjustments to the box to better match the object shape. Additionally, the network combines predictions from multiple feature maps with different resolutions to naturally handle objects of various sizes. Our SSD model is simple relative to methods that require object proposals because it completely eliminates proposal generation and subsequent pixel or feature resampling stage and encapsulates all computation in a single network. This makes SSD easy to train and straightforward to integrate into systems that require a detection component. Experimental results on the PASCAL VOC, MS COCO, and ILSVRC datasets confirm that SSD has comparable accuracy to methods that utilize an additional object proposal step and is much faster, while providing a unified framework for both training and inference. Compared to other single stage methods, SSD has much better accuracy, even with a smaller input image size.

**2.3. Faster R-CNN: towards real-time object detection with region proposal networks [C] Proceedings of the 2015 advances in Neural Information Processing Systems**

State-of-the-art object detection networks depend on region proposal algorithms to hypothesize object locations. Advances like SPPnet and Fast R-CNN have reduced the running time of these detection networks, exposing region proposal computation as a bottleneck. In this work, we introduce a Region Proposal Network (RPN) that shares full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals. An RPN is a fully-convolutional network that simultaneously predicts object bounds and objectness scores at each position. RPNs are trained end-to-end to generate high-quality region proposals, which are used by Fast R-CNN for detection. With a simple alternating optimization, RPN and Fast R-CNN can be trained to share convolutional features. For the very deep VGG-16 model, our detection system has a frame rate of 5fps (including all steps) on a GPU, while achieving state-of-the-art object detection accuracy on PASCAL VOC 2007 (73.2% mAP) and 2012 (70.4% mAP) using 300 proposals per image. Code is available at https://github.com/ShaoqingRen/faster\_rcnn.

**2.4.An Incremental Improvement [C]//IEEE Conference on Computer Vision and Pattern Recognition.**

We present some updates to YOLO! We made a bunch of little design changes to make it better. We also trained this new network that’s pretty swell. It’s a little bigger than last time but more accurate. It’s still fast though, don’t worry. At 320 × 320 YOLOv3 runs in 22 ms at 28.2 mAP, as accurate as SSD but three times faster. When we look at the old .5 IOU mAP detection metric YOLOv3 is quite good. It achieves 57.9 AP50 in 51 ms on a Titan X, compared to 57.5 AP50 in 198 ms by RetinaNet, similar performance but 3.8× faster.

**2.5. Densely connected convolutional networks[C]//IEEE Conference on Computer Vision and Pattern Recognition**

Recent work has shown that convolutional networks can be substantially deeper, more accurate, and efficient to train if they contain shorter connections between layers close to the input and those close to the output. In this paper, we embrace this observation and introduce the Dense Convolutional Network (DenseNet), which connects each layer to every other layer in a feed-forward fashion. Whereas traditional convolutional networks with L layers have L connections-one between each layer and its subsequent layer-our network has L(L+1)/2 direct connections. For each layer, the feature-maps of all preceding layers are used as inputs, and its own feature-maps are used as inputs into all subsequent layers. DenseNets have several compelling advantages: they alleviate the vanishing-gradient problem, strengthen feature propagation, encourage feature reuse, and substantially reduce the number of parameters. We evaluate our proposed architecture on four highly competitive object recognition benchmark tasks (CIFAR-10, CIFAR-100, SVHN.

**EXISTING SYSTEM**

Naturally four wheeler segments incorporated various safety features like ABS, Air bag and automated guidance system over the periods. But in two wheeler segments very little has been done to protect the two wheeler drivers by various manufactures. There were prototypes proposed with more features causing more components to be implanted on the helmet which can lead to add more weight to the helmet and also more cost involved in developing it. There was no discussion on the protection of human head when it exposed to severe accident which results in electronics are being embedded to the human head. There was no prototype available or designed with solar energy as power source to the smart helmet.

**DRAWBACK OF EXISTING SYSTEM**

* However, the detection model has a poor performance when the images are not very clear, the safety helmets are too small and obscure, and the background is too complex.
* the presented model is limited by the problems that some images of the dataset are less in quantity.
* the preprocessing operations of the images are confined to rotation, cutting, and zooming; the manual labeling is not comprehensive and may miss some objects.
* In some extreme cases, for example, only part of the head is visible and the safety helmet is obstructed, the model cannot detect the helmets accurately

**PROPOSED SYSTEM**

In this paper we approach a new method in deep learning(CNN) for prediction of either helmet wearing and gloves or not. Safety has always been a very important issue in all industrial activities, especially construction. It is after all, not a run-of-the-mill office job and requires some precautionary measures. The more prepared the Labourers/workers are, the less chance for accidents and injuries at a construction site. The head is the only organ of the human body that is totally encased in bone. This by decree of nature states the importance of protecting a very vital functioning component of our body, the brain. Hard hats or Safety helmets and glove act as the first line of defense against head injury, but they only work when they are worn correctly. Thus, it’s safe to say safety Helmets save lives and reduce the risk of brain injury A Convolutional Neural Network is a class of deep learning, feed-forward artificial neural networks, and most commonly useful for several analyses. A convolutional neural network (CNN) is a multilayer neural network. It is a deep learning method designed for image recognition and classification tasks. It can solve the problems of too many parameters and difficult training of the deep neural networks and can get better classification effects. The structure of most CNNs consists of input layer-convolutional layer (Conv layer)-activation function-pooling layer-fully connected layer (FC layer). The main characteristics of CNNs are local connectivity and parameter sharing in order to reduce the number of parameters and increase the efficiency of detection.

The Conv layer and the pooling layer are the core parts, and they can extract the object features. Often, the convolutional layer and the pooling layer may occur alternately. The Conv layers can extract and reinforce the object features. The pooling layers can filter multiple features, remove the unimportant features, and compress the features. The activation layers use nonlinear activation functions to enhance the expression ability of the neural network models and can solve the nonlinear problems effectively. The FC layers combine the data features of objects and output the feature values. By this means the CNNs can transfer the original input images from the original pixel values to the final classification confidence layer by layer.

In order to better extract the object features and classify the objects more precisely, proposed the concept of deep learning which is to learn object features from vast amounts of data using deep neural networks and then classify new objects according to the learned features. Deep learning algorithm based on convolutional neural networks has achieved great results in object detection, image recognition, and image segmentation. proposed R-CNN detection framework (region with CNN features) in 2014. Many models based on R-CNN were proposed after that including SPP-net (spatial pyramid pooling network) , Fast R-CNN (fast region with CNN features), and Faster R-CNN (faster region with CNN features) .

The R-CNN algorithm is based on a feed-forward convolutional network to produce bounding boxes of fixed sizes and generate scores for the object class examples in the boxes. A non maximum suppression method is used to predict the final results.

The early network layers of the SSD model are called the base network, based on a standard framework to classify the image. The base network is truncated before the classification layers, and the convolutional layers are added at the end of the truncated base network. The sizes of the convolutional feature maps decrease progressively to predict the detections at multiple scales.

The R-CNN algorithm sets a series of fixed and different size default boxes on the cell of each feature .Each default box predicts two kinds of detections. One is the location of bounding boxes including 4 offsets , which represent, respectively, *x* and *y* coordinates of the center of the bounding box and the width and height of the bounding box; the other is the score of each class. If there are *C* classes of the objects, the SSD algorithm predicts a total of *C*+1 score including the score of the background.

**ADVANTAGES OF PROPOSED SYSTEM**

* The proposed system automatic monitoring method can contribute to monitoring the construction workers and confirm the safety helmet wearing conditions at the construction site.
* Get workers’ continuous body vitals, reducing cases of accidents, and increasing quick response time.
* computer vision-based object detection, we develop a deep learning based method for the real-time detection of safety helmet at the construction site.

**BLOCK DIAGRAM**

Dataset, Train, Test

Dataset Pre-processing

Maxpool, Conv2D, Dense, Flatten

Faster R-CNN Model

Faster R-CNN Model

Predicted output

Testing with a new input image

The proposed system of binary classification will detect whether Air pollution is present or not. There are Training data, Test data and validation present in the proposed data model. With the help of training data, we just going to explain the system to identify the accurate results. In the proposed training process, the dataset is fed to the data preprocessing model. The data preprocessing model helps to eliminate the unwanted images and then given to that neural network layers (Conv2D,Maxpool,dense and flatten layers ) used to enhance the pixel quality by convolve 2 dimensional array and max pooling operation is performed on the received input which is identification of highest value in each patch of feature map, dense layer is used to classify image based on output from convolutional layers and finally flatten layer is used to make the multidimensional input into one dimensional flatten layer or fully connected layer. On successful completion of neural network layer process, automatically RCNN file will be generated. The CNN model will analyze the image given input image and predict the correct result using the data pre-trained. Thus the correct desired output is got from CNN model .The output classified using binary classification.

**CLASS DIAGRAM:**

**DATASET**

**TRAINING**

**TESTING**

**CNN ALGORITHM**

**TENSOR FLOW RECORD**

**CLASSIFICATION**

**NEW DATA**

**PREDICTED OUTCOME**

**TENSOR FLOW RECORD**

**PROPOSED METHODOLOGY**

A CNN is a neural network with a certain level of complexity, a neural network with more than two layers. Convolutional neural networks use sophisticated mathematical modelling to process data in complex ways. Convolutional neural networks as networks that have an input layer, an output layer and at least one hidden layer in between. Each layer performs specific types of sorting and ordering in a process that some refer to as “feature hierarchy.” One of the key uses of these sophisticated neural networks is dealing with unlabeled or unstructured data. The phrase “deep learning” is also used to describe these Convolutional neural networks, as deep learning represents a specific form of machine learning where technologies using aspects of artificial intelligence seek to classify and order information in ways that go beyond simple input/output protocols. The CNN finds the correct mathematical manipulation to turn the input into the output, whether it be as linear relationship or a non-linear relationship. The network moves through the layers calculating the probability of each output. CNN architectures generate compositional models where the object is expressed as a layered composition of primitives. The extra layers enable composition of features from lower layers, potentially modelling complex data with fewer units than a similarly performing shallow network.

**CONVOLUTIONAL NEURAL NETWORK**

In [deep learning](https://en.wikipedia.org/wiki/Deep_learning), a convolutional neural network (CNN, or ConvNet) is a class of [Artificial Neural Network](https://en.wikipedia.org/wiki/Artificial_Neural_Network)(ANN), most commonly applied to analyze visual imagery.[[1]](https://en.wikipedia.org/wiki/Convolutional_neural_network#cite_note-Valueva_Nagornov_Lyakhov_Valuev_2020_pp._232%E2%80%93243-1) They are also known as Shift Invariant or Space Invariant Artificial Neural Networks (SIANN), based on the shared-weight architecture of the convolution kernels or filters that slide along input features and provide translation [equivariant](https://en.wikipedia.org/wiki/Equivariant_map) responses known as feature maps. Counter-intuitively, most convolutional neural networks are only [equivariant](https://en.wikipedia.org/wiki/Equivariant_map), as opposed to [invariant](https://en.wikipedia.org/wiki/Translation_invariant), to translation. They have applications in [image and video recognition](https://en.wikipedia.org/wiki/Computer_vision), [recommender systems](https://en.wikipedia.org/wiki/Recommender_system), [image classification](https://en.wikipedia.org/wiki/Image_classification), [image segmentation](https://en.wikipedia.org/wiki/Image_segmentation), [medical image analysis](https://en.wikipedia.org/wiki/Medical_image_computing), [natural language processing](https://en.wikipedia.org/wiki/Natural_language_processing), [brain-computer interfaces](https://en.wikipedia.org/wiki/Brain%E2%80%93computer_interface), and financial [time series](https://en.wikipedia.org/wiki/Time_series).

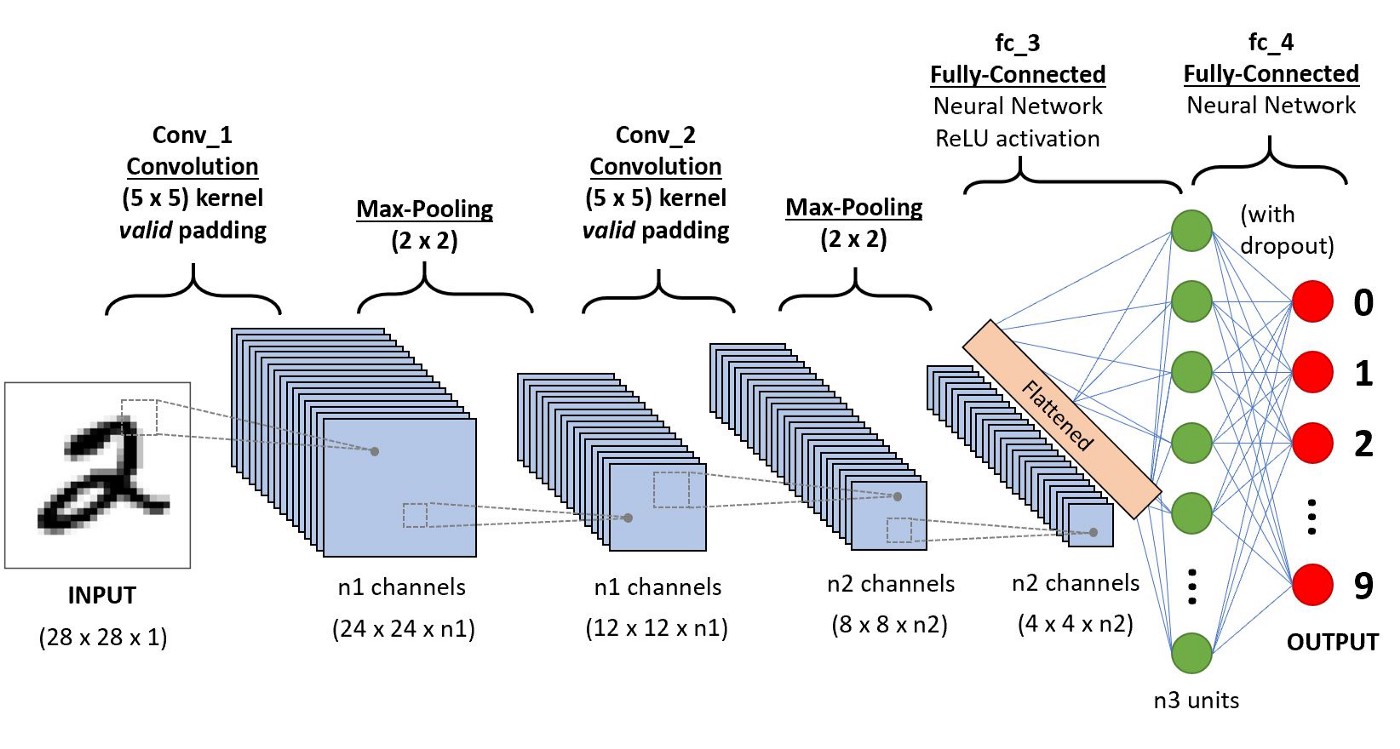
CNNs are [regularized](https://en.wikipedia.org/wiki/Regularization_(mathematics)) versions of [multilayer perceptrons](https://en.wikipedia.org/wiki/Multilayer_perceptron). Multilayer perceptrons usually mean fully connected networks, that is, each neuron in one [layer](https://en.wikipedia.org/wiki/Layer_(deep_learning)) is connected to all neurons in the next [layer](https://en.wikipedia.org/wiki/Layer_(deep_learning)). The "full connectivity" of these networks make them prone to [overfitting](https://en.wikipedia.org/wiki/Overfitting) data. Typical ways of regularization, or preventing overfitting, include: penalizing parameters during training (such as weight decay) or trimming connectivity (skipped connections, dropout, etc.) CNNs take a different approach towards regularization: they take advantage of the hierarchical pattern in data and assemble patterns of increasing complexity using smaller and simpler patterns embossed in their filters. Therefore, on a scale of connectivity and complexity, CNNs are on the lower extreme.

Convolutional networks were [inspired](https://en.wikipedia.org/wiki/Mathematical_biology) by [biological](https://en.wikipedia.org/wiki/Biological) processes in that the connectivity pattern between [neurons](https://en.wikipedia.org/wiki/Artificial_neuron) resembles the organization of the animal [visual cortex](https://en.wikipedia.org/wiki/Visual_cortex). Individual [cortical neurons](https://en.wikipedia.org/wiki/Cortical_neuron) respond to stimuli only in a restricted region of the [visual field](https://en.wikipedia.org/wiki/Visual_field) known as the [receptive field](https://en.wikipedia.org/wiki/Receptive_field). The receptive fields of different neurons partially overlap such that they cover the entire visual field. CNNs use relatively little pre-processing compared to other [image classification algorithms](https://en.wikipedia.org/wiki/Image_classification). This means that the network learns to optimize the [filters](https://en.wikipedia.org/wiki/Filter_(signal_processing)) (or kernels) through automated learning, whereas in traditional algorithms these filters are [hand-engineered](https://en.wikipedia.org/wiki/Feature_engineering). This independence from prior knowledge and human intervention in feature extraction is a major advantage.

**CONVOUTIONAL LAYERS**

In a CNN, the input is a [tensor](https://en.wikipedia.org/wiki/Tensor) with a shape: (number of inputs) x (input height) x (input width) x (input [channels](https://en.wikipedia.org/wiki/Channel_(digital_image))). After passing through a convolutional layer, the image becomes abstracted to a feature map, also called an activation map, with shape: (number of inputs) x (feature map height) x (feature map width) x (feature map [channels](https://en.wikipedia.org/wiki/Channel_(digital_image))).

Convolutional layers convolve the input and pass its result to the next layer. This is similar to the response of a neuron in the visual cortex to a specific stimulus.[[14]](https://en.wikipedia.org/wiki/Convolutional_neural_network#cite_note-deeplearning-14) Each convolutional neuron processes data only for its [receptive field](https://en.wikipedia.org/wiki/Receptive_field). Although [fully connected feed forward neural networks](https://en.wikipedia.org/wiki/Multilayer_perceptron) can be used to learn features and classify data, this architecture is generally impractical for larger inputs such as high resolution images. It would require a very high number of neurons, even in a shallow architecture, due to the large input size of images, where each pixel is a relevant input feature. For instance, a fully connected layer for a (small) image of size 100 x 100 has 10,000 weights for *each* neuron in the second layer. Instead, convolution reduces the number of free parameters, allowing the network to be deeper.[[15]](https://en.wikipedia.org/wiki/Convolutional_neural_network#cite_note-15) For example, regardless of image size, using a 5 x 5 tiling region, each with the same shared weights, requires only 25 learnable parameters. Using regularized weights over fewer parameters avoids the vanishing gradients and exploding gradients problems seen during [back propagation](https://en.wikipedia.org/wiki/Backpropagation) in traditional neural networks. Furthermore, convolutional neural networks are ideal for data with a grid-like topology (such as images) as spatial relations between separate features are taken into account during convolution and/or pooling.



**Layers in CNN**

There are five different layers in CNN

Input layer

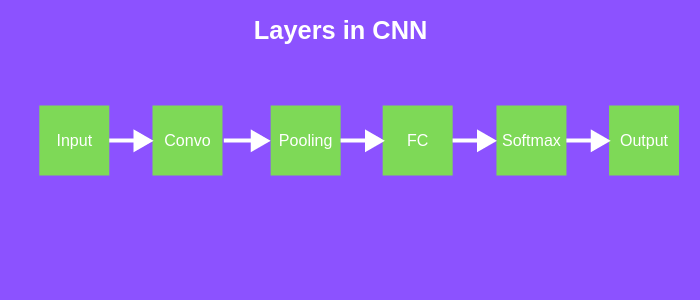
Convo layer (Convo + ReLU)

Pooling layer

Fully connected(FC) layer

Softmax/logistic layer

Output layer



**Different layers of CNN**

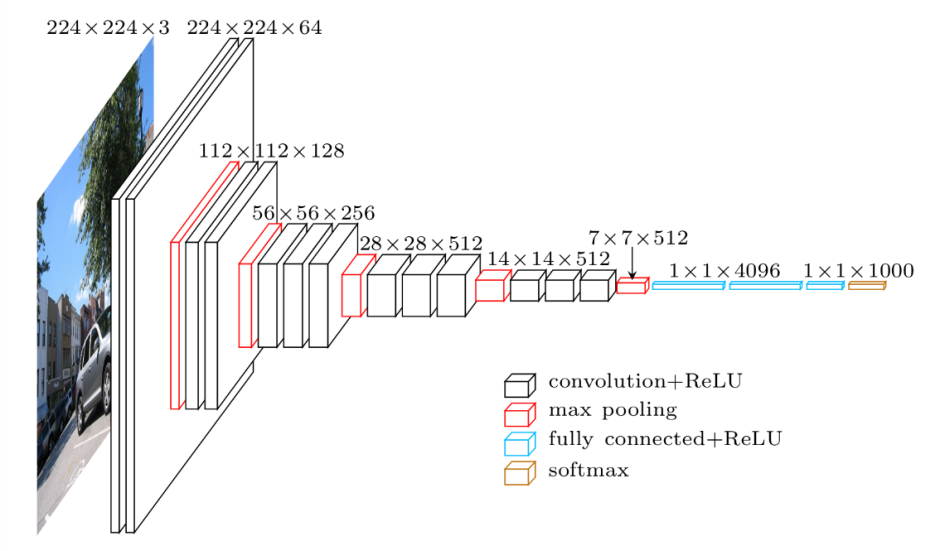
**Input Layer**

Input layer in CNN should contain image data. Image data is represented by three dimensional matrix as we saw earlier. You need to reshape it into a single column. Suppose you have image of dimension 28 x 28 =784, you need to convert it into 784 x 1 before feeding into input. If you have “m” training examples then dimension of input will be (784, m).

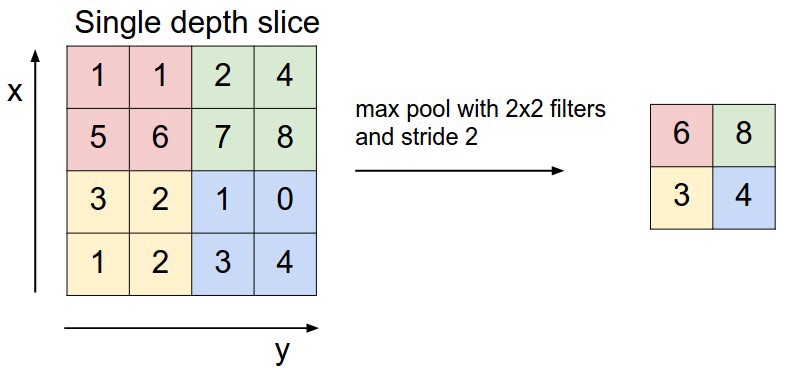
**Convo Layer**

Convo layer is sometimes called feature extractor layer because features of the image are get extracted within this layer. First of all, a part of image is connected to Convo layer to perform convolution operation as we saw earlier and calculating the dot product between receptive field (it is a local region of the input image that has the same size as that of filter) and the filter. Result of the operation is single integer of the output volume. Then we slide the filter over the next receptive field of the same input image by a Stride and do the same operation again. We will repeat the same process again and again until we go through the whole image. The output will be the input for the next layer.

Convo layer also contains ReLU activation to make all negative value to zero.



**Pooling Layer**



Pooling layer is used to reduce the spatial volume of input image after convolution. It is used between two convolution layer. If we apply FC after Convo layer without applying pooling or max pooling, then it will be computationally expensive and we don’t want it. So, the max pooling is only way to reduce the spatial volume of input image. In the above example, we have applied max pooling in single depth slice with Stride of 2. You can observe the 4 x 4 dimension input is reduce to 2 x 2 dimension.

There is no parameter in pooling layer but it has two hyperparameters — Filter(F) and Stride(S).

In general, if we have input dimension W1 x H1 x D1, then

W2 = (W1−F)/S+1

H2 = (H1−F)/S+1

D2 = D1

Where W2, H2 and D2 are the width, height and depth of output.

**Fully Connected Layer (FC)**

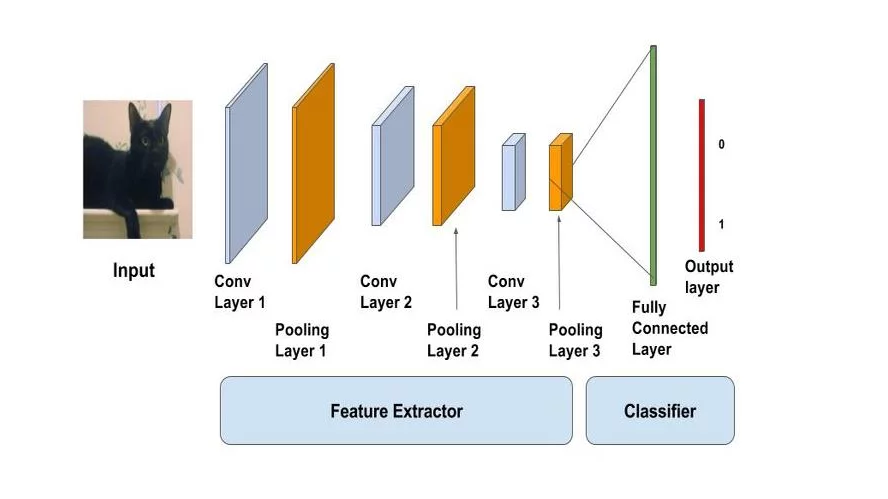
Fully connected layer involves weights, biases, and neurons. It connects neurons in one layer to neurons in another layer. It is used to classify images between different categories by training.

**Softmax / Logistic Layer**

**Softmax** or Logistic layer is the last layer of CNN. It resides at the end of FC layer. Logistic is used for binary classification and softmax is for multi-classification.

**Output Layer**

Output layer contains the label which is in the form of one-hot encoded.



**SOFTWARE DISCRIPTION**

* **PYTHON 3.7.3**
* **KERAS**
* **TENSORFLOW**
* **NUMPY**
* **PILLOW**

**PYTHON**

Python is a wonderful and powerful programming language that's easy to use (easy to read **and** write) and with Raspberry Pi lets you connect your project to the real world.

Python syntax is very clean, with an emphasis on readability and uses standard English keywords. Start by opening IDLE from the desktop.

**IDLE**

The easiest introduction to Python is through IDLE, a Python development environment. Open IDLE from the Desktop or applications menu:

IDLE gives you a REPL (Read-Evaluate-Print-Loop) which is a prompt you can enter Python commands in to. As it's a REPL you even get the output of commands printed to the screen without using print.

>>> 1 + 2

3

>>>name = "Sarah"

>>> "Hello " + name

'Hello Sarah'

IDLE also has syntax highlighting built in and some support for autocompletion. You can look back on the history of the commands you've entered in the REPL with Alt + P (previous) and Alt + N (next).

**BASIC PYTHON USAGE**

Hello world in Python:

print("Hello world")

Simple as that!

**INDENTATION**

Some languages use curly braces { and } to wrap around lines of code which belong together, and leave it to the writer to indent these lines to appear visually nested. However, Python does not use curly braces but instead requires indentation for nesting. For example a for loop in Python:

fori in range(10):

print("Hello")

The indentation is necessary here. A second line indented would be a part of the loop, and a second line not indented would be outside of the loop. For example:

fori in range(2):

print("A")

print("B")

would print:

A

B

A

B

whereas the following:

fori in range(2):

print("A")

print("B")

would print:

A

A

B

**VARIABLES**

To save a value to a variable, assign it like so:

name = "Bob"

age = 15

Note here I did not assign types to these variables, as types are inferred, and can be changed (it's dynamic).

age = 15

age += 1 # increment age by 1

print(age)

This time I used comments beside the increment command.

**COMMENTS**

Comments are ignored in the program but there for you to leave notes, and are denoted by the hash # symbol. Multi-line comments use triple quotes like so:

"""

This is a very simple Python program that prints "Hello".

That's all it does.

"""

print("Hello")

**LISTS**

Python also has lists (called arrays in some languages) which are collections of data of any type:

numbers = [1, 2, 3]

Lists are denoted by the use of square brackets [] and each item is separated by a comma.

**ITERATION**

Some data types are iterable, which means you can loop over the values they contain. For example a list:

numbers = [1, 2, 3]

for number in numbers:

print(number)

This takes each item in the list numbers and prints out the item:

1

2

3

Note I used the word number to denote each item. This is merely the word I chose for this - it's recommended you choose descriptive words for variables - using plurals for lists, and singular for each item makes sense. It makes it easier to understand when reading.

Other data types are iterable, for example the string:

dog\_name = "BINGO"

for char in dog\_name:

print(char)

This loops over each character and prints them out:

B

I

N

G

O

**RANGE**

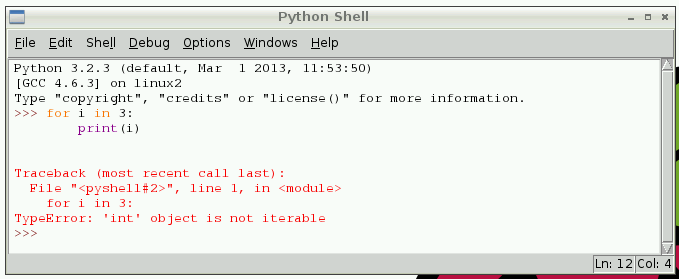
The integer data type is not iterable and trying to iterate over it will produce an error. For example:

fori in 3:

print(i)

will produce:

TypeError: 'int' object is not iterable



However you can make an iterable object using the range function:

fori in range(3):

print(i)

range(5) contains the numbers 0, 1, 2, 3 and 4 (five numbers in total). To get the numbers 1 to 5 use range(1, 6).

**LENGTH**

You can use functions like len to find the length of a string or a list:

name = "Jamie"

print(len(name)) # 5

names = ["Bob", "Jane", "James", "Alice"]

print(len(names)) # 4

**IF STATEMENTS**

You can use if statements for control flow:

name = "Joe"

iflen(name) > 3:

print("Nice name,")

print(name)

else:

print("That's a short name,")

print(name)

**PYTHON FILES IN IDLE**

To create a Python file in IDLE, click File > New File and you'll be given a blank window. This is an empty file, not a Python prompt. You write a Python file in this window, save it, then run it and you'll see the output in the other window.

For example, in the new window, type:

n = 0

fori in range(1, 101):

n += i

print("The sum of the numbers 1 to 100 is:")

print(n)

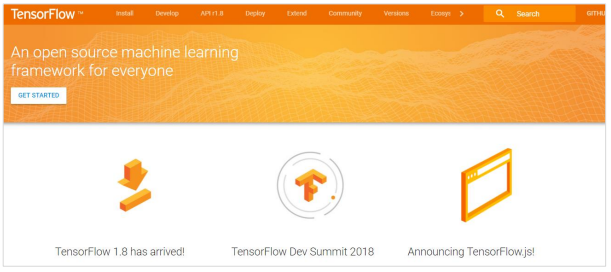
Then save this file (File > Save or Ctrl + S) and run (Run > Run Moduleor hit F5) and you'll see the output in your original Python window.

**EXECUTING PYTHON FILES FROM THE COMMAND LINE**

You can write a Python file in a standard [editor](https://www.raspberrypi.org/documentation/linux/usage/text-editors.md) like Vim, Nano or LeafPad, and run it as a Python script from the command line. Just navigate to the directory the file is saved (use cd and ls for guidance) and run with python, e.g. python hello.py.

**TENSORFLOW-** **INTRODUCTION**

TensorFlow is a software library or framework, designed by the Google team to implement machine learning and deep learning concepts in the easiest manner. It combines the computational algebra of optimization techniques for easy calculation of many mathematical expressions. The official website of TensorFlow is mentioned below: <https://www.tensorflow.org/>



Let us now consider the following important features of TensorFlow:

• It includes a feature of that defines, optimizes and calculates mathematical expressions easily with the help of multi-dimensional arrays called tensors.

• It includes a programming support of deep neural networks and machine learning techniques.

• It includes a high scalable feature of computation with various data sets.

• TensorFlow uses GPU computing, automating management. It also includes a unique feature of optimization of same memory and the data used.

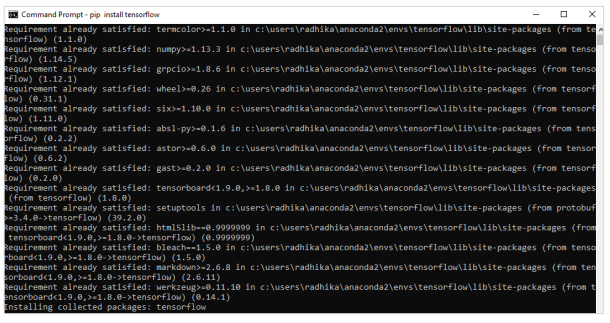
**Why is TensorFlow So Popular?**

TensorFlow is well-documented and includes plenty of machine learning libraries. It offers a few important functionalities and methods for the same. TensorFlow is also called a “Google” product. It includes a variety of machine learning and deep learning algorithms. TensorFlow can train and run deep neural networks for handwritten digit classification, image recognition, word embedding and creation of various sequence models.

**TensorFlow — Installation**

To install TensorFlow, it is important to have “Python” installed in your system. Python version 3.4+ is considered the best to start with TensorFlow installation. Consider the following steps to install TensorFlow in Windows operating system.

**pip install tensorflow**

****

**TensorFlow — Convolutional Neural Networks**

After understanding machine-learning concepts, we can now shift our focus to deep learning concepts. Deep learning is a division of machine learning and is considered as a crucial step taken by researchers in recent decades. The examples of deep learning implementation include applications like image recognition and speech recognition.

Following are the two important types of deep neural networks:

• Convolutional Neural Networks

• Recurrent Neural Networks In this chapter, we will focus on the CNN, Convolutional Neural Networks

**Convolutional Neural Networks**

Convolutional Neural networks are designed to process data through multiple layers of arrays. This type of neural networks is used in applications like image recognition or face recognition. The primary difference between CNN and any other ordinary neural network is that CNN takes input as a two-dimensional array and operates directly on the images rather than focusing on feature extraction which other neural networks focus on. The dominant approach of CNN includes solutions for problems of recognition. Top companies like Google and Facebook have invested in research and development towards recognition projects to get activities done with greater speed.

A convolutional neural network uses three basic ideas:

• Local respective fields

• Convolution

• Pooling

Let us understand these ideas in detail.

CNN utilizes spatial correlations that exist within the input data. Each concurrent layer of a neural network connects some input neurons. This specific region is called local receptive field. Local receptive field focusses on the hidden neurons. The hidden neurons process the input data inside the mentioned field not realizing the changes outside the specific boundary.

If we observe the above representation, each connection learns a weight of the hidden neuron with an associated connection with movement from one layer to another. Here, individual neurons perform a shift from time to time. This process is called “convolution”. The mapping of connections from the input layer to the hidden feature map is defined as “shared weights” and bias included is called “shared bias”. CNN or convolutional neural networks use pooling layers, which are the layers, positioned immediately after CNN declaration. It takes the input from the user as a feature map that comes out of convolutional networks and prepares a condensed feature map. Pooling layers helps in creating layers with neurons of previous layers.

**KERAS**

**INTRODUCTION**

Deep learning is one of the major subfield of machine learning framework. Machine learning is the study of design of algorithms, inspired from the model of human brain. Deep learning is becoming more popular in data science fields like robotics, artificial intelligence(AI), audio & video recognition and image recognition. Artificial neural network is the core of deep learning methodologies. Deep learning is supported by various libraries such as Theano, TensorFlow, Caffe, Mxnet etc., Keras is one of the most powerful and easy to use python library, which is built on top of popular deep learning libraries like TensorFlow, Theano, etc., for creating deep learning models.

**OVERVIEW OF KERAS**

Keras runs on top of open source machine libraries like TensorFlow, Theano or Cognitive Toolkit (CNTK). Theano is a python library used for fast numerical computation tasks. TensorFlow is the most famous symbolic math library used for creating neural networks and deep learning models. TensorFlow is very flexible and the primary benefit is distributed computing. CNTK is deep learning framework developed by Microsoft. It uses libraries such as Python, C#, C++ or standalone machine learning toolkits. Theano and TensorFlow are very powerful libraries but difficult to understand for creating neural networks. Keras is based on minimal structure that provides a clean and easy way to create deep learning models based on TensorFlow or Theano. Keras is designed to quickly define deep learning models. Well, Keras is an optimal choice for deep learning applications.

**FEATURES**

Keras leverages various optimization techniques to make high level neural network API easier and more performant. It supports the following features:

• Consistent, simple and extensible API.

• Minimal structure - easy to achieve the result without any frills.

• It supports multiple platforms and backends.

• It is user friendly framework which runs on both CPU and GPU.

• Highly scalability of computation.

**BENEFITS**

Keras is highly powerful and dynamic framework and comes up with the following advantages:

• Larger community support.

• Easy to test.

• Keras neural networks are written in Python which makes things simpler.

• Keras supports both convolution and recurrent networks.

• Deep learning models are discrete components, so that, you can combine into many ways.

**KERAS ― OVERVIEW OF DEEP LEARNING**

Deep learning is an evolving subfield of machine learning. Deep learning involves analyzing the input in layer by layer manner, where each layer progressively extracts higher level information about the input. Let us take a simple scenario of analyzing an image. Let us assume that your input image is divided up into a rectangular grid of pixels. Now, the first layer abstracts the pixels. The second layer understands the edges in the image. The Next layer constructs nodes from the edges. Then, the next would find branches from the nodes. Finally, the output layer will detect the full object. Here, the feature extraction process goes from the output of one layer into the input of the next subsequent layer. By using this approach, we can process huge amount of features, which makes deep learning a very powerful tool. Deep learning algorithms are also useful for the analysis of unstructured data. Let us go through the basics of deep learning in this chapter.

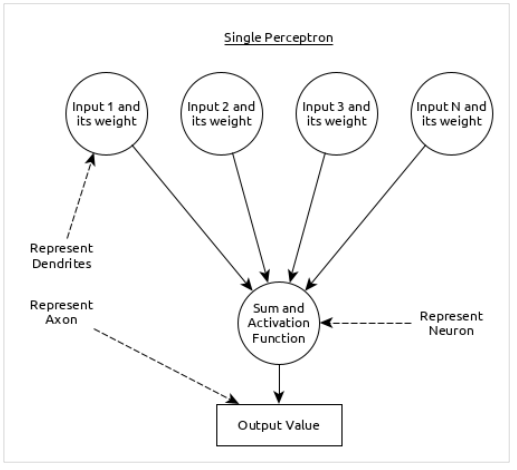
**Artificial Neural Networks**

The most popular and primary approach of deep learning is using “Artificial neural network” (ANN). They are inspired from the model of human brain, which is the most complex organ of our body. The human brain is made up of more than 90 billion tiny cells called “Neurons”. Neurons are inter-connected through nerve fiber called “axons” and “Dendrites”. The main role of axon is to transmit information from one neuron to another to which it is connected.

Similarly, the main role of dendrites is to receive the information being transmitted by the axons of another neuron to which it is connected. Each neuron processes a small information and then passes the result to another neuron and this process continues. This is the basic method used by our human brain to process huge about of information like speech, visual, etc., and extract useful information from it.

Based on this model, the first Artificial Neural Network (ANN) was invented by psychologist Frank Rosenblatt, in the year of 1958. ANNs are made up of multiple nodes which is similar to neurons. Nodes are tightly interconnected and organized into different hidden layers. The input layer receives the input data and the data goes through one or more hidden layers sequentially and finally the output layer predict something useful about the input data. For example, the input may be an image and the output may be the thing identified in the image, say a “Cat”.

A single neuron (called as perceptron in ANN) can be represented as below:

****

Here,

• Multiple input along with weight represents dendrites.

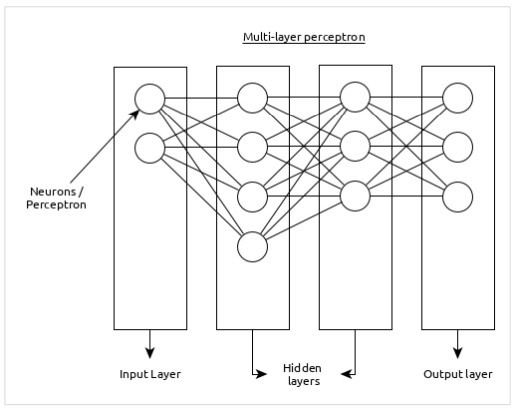
• Sum of input along with activation function represents neurons. Sum actually means computed value of all inputs and activation function represent a function, which modify the Sum value into 0, 1 or 0 to 1.

• Actual output represent axon and the output will be received by neuron in next layer.

Let us understand different types of artificial neural networks in this section.

**Multi-Layer Perceptron**

Multi-Layer perceptron is the simplest form of ANN. It consists of a single input layer, one or more hidden layer and finally an output layer. A layer consists of a collection of perceptron. Input layer is basically one or more features of the input data. Every hidden layer consists of one or more neurons and process certain aspect of the feature and send the processed information into the next hidden layer. The output layer process receives the data from last hidden layer and finally output the result.

****

**Convolutional Neural Network (CNN)**

Convolutional neural network is one of the most popular ANN. It is widely used in the fields of image and video recognition. It is based on the concept of convolution, a mathematical concept. It is almost similar to multi-layer perceptron except it contains series of convolution layer and pooling layer before the fully connected hidden neuron layer.

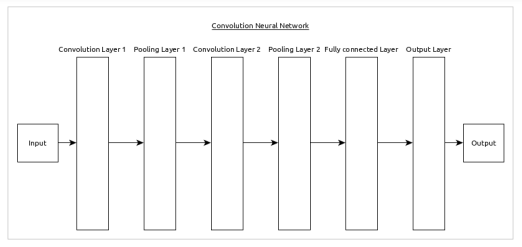
It has three important layers:

• Convolution layer: It is the primary building block and perform computational tasks based on convolution function.

• Pooling layer: It is arranged next to convolution layer and is used to reduce the size of inputs by removing unnecessary information so computation can be performed faster.

• Fully connected layer: It is arranged to next to series of convolution and pooling layer and classify input into various categories.

A simple CNN can be represented as below:



Here,

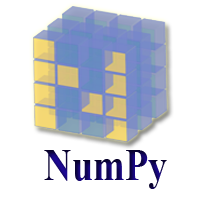
• 2 series of Convolution and pooling layer is used and it receives and process the input (e.g. image).

• A single fully connected layer is used and it is used to output the data (e.g. classification of image)

**PYTHON NUMPY**

Our Python NumPy Tutorial provides the basic and advanced concepts of the NumPy. Our NumPy tutorial is designed for beginners and professionals.

NumPy stands for numeric python which is a python package for the computation and processing of the multidimensional and single dimensional array elements.



**What is NumPy**

NumPy stands for numeric python which is a python package for the computation and processing of the multidimensional and single dimensional array elements.

Travis Oliphant created NumPy package in 2005 by injecting the features of the ancestor module Numeric into another module Numarray. It is an extension module of Python which is mostly written in C. It provides various functions which are capable of performing the numeric computations with a high speed.

NumPy provides various powerful data structures, implementing multi-dimensional arrays and matrices. These data structures are used for the optimal computations regarding arrays and matrices. In this tutorial, we will go through the numeric python library NumPy.

**The need of NumPy**

With the revolution of data science, data analysis libraries like NumPy, SciPy, Pandas, etc. have seen a lot of growth. With a much easier syntax than other programming languages, python is the first choice language for the data scientist.

NumPy provides a convenient and efficient way to handle the vast amount of data. NumPy is also very convenient with Matrix multiplication and data reshaping. NumPy is fast which makes it reasonable to work with a large set of data.

There are the following advantages of using NumPy for data analysis.

1. NumPy performs array-oriented computing.
2. It efficiently implements the multidimensional arrays.
3. It performs scientific computations.
4. It is capable of performing Fourier Transform and reshaping the data stored in multidimensional arrays.
5. NumPy provides the in-built functions for linear algebra and random number generation.

Nowadays, NumPy in combination with SciPy and Mat-plotlib is used as the replacement to MATLAB as Python is more complete and easier programming language than MATLAB.

**Prerequisite**

Before learning Python Numpy, you must have the basic knowledge of Python concepts.

**PYTHON PILLOW ― OVERVIEW**

In today’s digital world, we come across lots of digital images. In case, we are working with Python programming language, it provides lot of image processing libraries to add image processing capabilities to digital images.

Some of the most common image processing libraries are: OpenCV, Python Imaging Library (PIL), Scikit-image, Pillow. However, in this tutorial, we are only focusing on Pillow module and will try to explore various capabilities of this module.

Pillow is built on top of PIL (Python Image Library). PIL is one of the important modules for image processing in Python. However, the PIL module is not supported since 2011 and doesn’t support python 3.

Pillow module gives more functionalities, runs on all major operating system and support for python 3. It supports wide variety of images such as “jpeg”, “png”, “bmp”, “gif”, “ppm”, “tiff”. You can do almost anything on digital images using pillow module. Apart from basic image processing functionality, including point operations, filtering images using built-in convolution kernels, and color space conversions.

**IMAGE ARCHIVES**

The Python Imaging Library is best suited for image archival and batch processing applications. Python pillow package can be used for creating thumbnails, converting from one format to another and print images, etc.

**IMAGE DISPLAY**

You can display images using Tk PhotoImage, BitmapImage and Windows DIB interface, which can be used with PythonWin and other Windows-based toolkits and many other Graphical User Interface (GUI) toolkits.

For debugging purposes, there is a show () method to save the image to disk which calls the external display utility.

**Image Processing**

The Pillow library contains all the basic image processing functionality. You can do image resizing, rotation and transformation.

Pillow module allows you to pull some statistics data out of image using histogram method, which later can be used for statistical analysis and automatic contrast enhancement.

**PYTHON PILLOW — ENVIRONMENT SETUP**

This chapter discusses how to install pillow package in your computer.

Installing pillow package is very easy, especially if you’re installing it using pip.

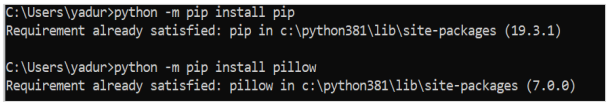
**Installing Pillow using pip**

To install pillow using pip, just run the below command in your command prompt:

python -m pip install pip

python -m pip install pillow

In case, if pip and pillow are already installed in your computer, above commands will simply mention the ‘requirement already satisfied’ as shown below:

****

**PYTHON PILLOW — USING IMAGE MODULE**

To display the image, pillow library is using an image class within it. The image module inside pillow package contains some important inbuilt functions like, load images or create new images, etc.

**Opening, rotating anddisplaying an image**

To load the image, we simply import the image module from the pillow and call the Image.open(), passing the image filename.

Instead of calling the Pillow module, we will call the PIL module as to make it backward compatible with an older module called Python Imaging Library (PIL). That’s why our code starts with “from PIL import Image” instead of “from Pillow import Image”.

Next, we’re going to load the image by calling the Image.open() function, which returns a value of the Image object data type. Any modification we make to the image object can be saved to an image file with the save() method. The image object we received using Image.open(), later can be used to resize, crop, draw or other image manipulation method calls on this Image object.

**Example**

Following example demonstrates the rotation of an image using python pillow:

from PIL import Image

#Open image using Image module

im = Image.open("images/cuba.jpg")

#Show actual Image im.show()

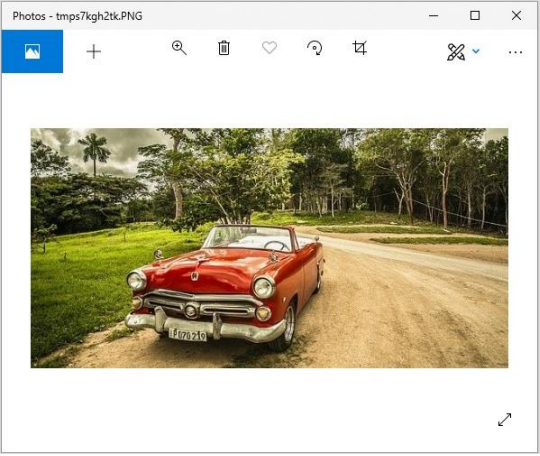
#Show rotated Image

im = im.rotate(45) im.show()

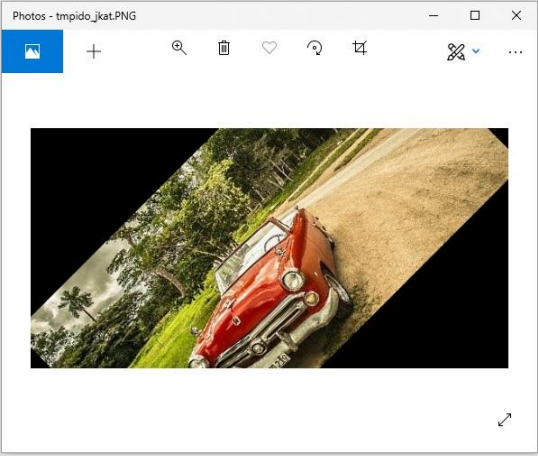
**Output**

If you save the above program as Example.py and execute, it displays the original and rotated images using standard PNG display utility, as follows:

**ACTUAL IMAGE**

****

**Rotated image (45 degrees)**

****

**RESULTS**

In the paper, the open-source Tensor Flow framework is chosen to train the model. The pre trained SSD\_mobilenet\_v1\_Kaggle model with the kaggle dataset is used to learn the characteristics of the safety helmet in the built dataset to reduce the training time and save the computing resources. The initial weights and the parameter values of our own model are the same as the SSD\_mobilenet\_v1\_kaggle model. Finally, the weights and the parameter values of the safety helmet detection model are trained and obtained through the training process.

Among the 3261 images, 2769 images were divided into the training set, 339 images were divided into the validation set, and 153 images were divided into the test set. The training set is used to train the model or to determine the parameters of the model. The validation set is used to adjust the hyper parameters of the model and to evaluate the capacity of the model preliminarily. The test set is used to evaluate the generalization ability of the final model [28].

In the course of training, the change of the mean average precision (mAP) and the loss function during training was recorded by Tensor Board. As a measure index, the mean average precision (mAP) is generally used in the field of object detection. Figure [6](https://www.hindawi.com/journals/ace/2020/9703560/fig6/) illustrated that the mean Average Precision shows an overall upward trend, and the trend has ups and downs and is not a steady rise. When training rounds up to 50,000, the mean average precision of the detection model is 36.82%. that the total loss values decrease slowly at the beginning of the training and converge at the end of the training. The values of the loss function are the differences between the true value and the predicted value in general speaking. The change in the values of the loss function represents the training process of the model.



The precision and recall are the commonly used metrics to evaluate the performance and reliability of the trained model. The precision of the trained model is 95% and the recall is 77%, which demonstrates that the proposed method performs well in safety helmet detection.

As the pictures above show, the probabilities of recognizing the safety helmets worn by workers as safety helmets are more than 80%. However, the output images of the model demonstrate some errors in the detection model. For example, it is hard for the model to detect the safety helmets of small sizes or large rotation angles. It is possible to recognize the objects of the same colors in the images as the safety helmets. When the illumination intensity of the construction site in the images and the objects are not clear, the safety helmets are difficult to be recognized. That suggests the detection model established in the paper is not accurate enough.

the probability predicted by the model is 98%, but the probability of recognizing the background as safety helmets is 78%. This fake detection generates false positive. This is a case of false detection which predicted the false object as correct. The case of Figure [10(c)](https://www.hindawi.com/journals/ace/2020/9703560/fig10/#c) is the same as the first one. the red helmet is missed and this is a case of false negative. The errors occur because of the interference of the complex background, the limitation of the number of the image dataset, and the safety helmets proportion in the images. In order to improve the performance of the model, some measures must be taken such as increasing the number of the image dataset and adding the preprocessing operations of the images. Besides the above measures, ameliorating the non-maximum suppression algorithm, adjusting the parameters and weights, and so forth can also be a great solution to reduce the false positives.

**CONCLUSIONS**

The paper proposed a method for detecting the wearing of safety helmets by the workers based on convolutional neural networks. The model uses the Faster RCNN-algorithm to detect safety helmets. Then, a dataset of 3261 images containing various helmets is built and divided into three parts to train and test the model. The Tensor Flow framework is chosen to train the model. After the training and testing process, the mean average precision (mAP) of the detection model is stable and the helmet detection model is built. The experiment results demonstrate that the method can be used to detect the safety helmets worn by the construction workers at the construction site. The presented method offers an alternative solution to detect the safety helmets and improve the safety management of the construction workers at the construction site.

#### References

1. X. Chang and X. M. Liu, “Fault tree analysis of unreasonably wearing helmets for builders,” *Journal of Jilin Jianzhu University*,
2. Z. Y. Wang, *Design and Implementation of Detection System of Warning Helmets Based on Intelligent Video Surveillance*, Beijing University of Posts and Telecommunications, Beijing, China, 2018.
3. H. Zeng, *Research on Intelligent Helmets System for Engineering Construction*, Harbin Institute of Technology, Harbin, China, 2017.
4. A. Kelm, L. Laußat, A. Meins-Becker et al., “Mobile passive radio frequency identification (RFID) portal for automated and rapid control of personal protective equipment (PPE) on construction sites,” *Automation in Construction*, vol. 36, pp. 38–52, 2013.
5. S. Barro-Torres, T. M. Fernández-Caramés, H. J. Pérez-Iglesias, and C. J. Escudero, “Real-time personal protective equipment monitoring system,” *Computer Communications*, vol. 36, no. 1, pp. 42–50, 2012.
6. A. H. M. Rubaiyat, T. T. Toma, M. Kalantari-Khandani et al., “Automatic detection of helmet uses for construction safety,” in *Proceedings of the 2016 IEEE ACM International Conference on Web Intelligence Workshops (WIW)*, ACM, Omaha, NE, USA,
7. K. Shrestha, P. P. Shrestha, D. Bajracharya, and E. A. Yfantis, “Hard-hat detection for construction safety visualization,” *Journal of Construction Engineering*, vol. 2015, Article ID 721380, 8 pages, 2015.View
8. R. Waranusast, N. Bundon, V. Timtong, C. Tangnoi, and P. Pattanathaburt, “Machine vision techniques for motorcycle safety helmet detection,” in *Proceedings of the Image & Vision Computing New Zealand*, IEEE, Wellington, New Zealand, November 2013.
9. P. Doungmala and K. Klubsuwan, “Helmet wearing detection in Thailand using haar like feature and circle hough transform on image processing,” in *Proceedings of the IEEE International Conference on Computer & Information Technology*, IEEE, Nadi, Fiji, December 2016.
10. J. S. Jia, Q. J. Bao, and H. M. Tang, “Method for detecting safety helmet based on deformable part model,” *Application Research of Computers*, vol. 33, no. 3, pp. 953–956, 2016.
11. K. Li, X. G. Zhao, J. Bian, and M. Tan, “Automatic safety helmet wearing detection,” 2018
12. H. Wu and J. Zhao, “Automated visual helmet identification based on deep convolutional neural networks,” in *Proceedings of the 13th International Symposium on Process Systems Engineering (PSE 2018)*, vol. 44, pp. 2299–2304, San Diego, CA, USA, July 2018.
13. S. Xu, Y. Wang, Y. Gu, N. Li, L. Zhuang, and L. Shi, “Safety helmet wearing detection study based on improved faster RCNN,” *Application Research of Computers*,
14. L. Ding, W. Fang, H. Luo, P. E. D. Love, B. Zhong, and X. Ouyang, “A deep hybrid learning model to detect unsafe behavior: integrating convolution neural networks and long short-term memory,” *Automation in Construction*, vol. 86,
15. Q. Fang, H. Li, X. Luo et al., “A deep learning-based method for detecting non-certified work on construction sites,” *Advanced Engineering Informatics*, vol. 35, pp. 56–68, 2018
16. W. Fang, L. Ding, H. Luo, and P. E. D. Love, “Falls from heights: a computer vision-based approach for safety harness detection,” *Automation in Construction*, vol. 91, pp. 53–61,
17. W. Fang, B. Zhong, N. Zhao et al., “A deep learning based approach for mitigating falls from height with computer vision: convolutional neural network,” *Advanced Engineering Informatics*, vol. 39, pp. 179–177, 2019.
18. H. Luo, C. Xiong, W. Fang, P. E. D. Love, B. Zhang, and X. Ouyang, “Convolutional neural networks: computer vision-based workforce activity assessment in construction,” *Automation in Construction*, vol. 94, pp. 282–289, 2018.
19. G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
20. R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, Columbus, OH, USA, June 2014.